**Sr. AZURE /INFRA /DEVOPS ENGINEER**

|  |  |
| --- | --- |
| **Harish G****mukesh@cosqube.com****703-223-8696** | **SFU_CRT_BDG_Admin_RGB2021-03_Badge_SF-Certified_Platform-Developer-I_500x490px2021-03_Badge_SF-Certified_Platform-Developer-I_500x490px2021-03_Badge_SF-Certified_Platform-Developer-I_500x490px2021-03_Badge_SF-Certified_Platform-Developer-I_500x490px2021-03_Badge_SF-Certified_Platform-Developer-I_500x490px2021-03_Badge_SF-Certified_Platform-Developer-I_500x490px** |

**SUMMARY:**

* DevOps automation engineer &**CI/CD** (Continuous Integration/Continuous Deployment) engineer with **10+** years of solid experience in working on **DevOps/Agile** operations process and as Developer around tools (Code review, unit test automation, Build & Release automation, Environment Management).
* Well Versed in building and deploying applications on to different environment such a **QA, UAT** and productions.
* Experience in various **Azure** Services like GoLand, Python, Compute (Web Roles, Worker Roles), Caching, **SQL Azure**, NoSQL, Storage and Network Services, Azure Active Directory, Internet of Things (IoT), Azure Monitoring, API **Management, Scheduling**, Auto scaling, Azure Identity & Access Management, Data Factory, PowerShell Automation and other **Azure Services and  Helm chart creation**.
* Experience in provisioning of **IaaS, PaaS** Virtual Machines and Web, Worker roles on **Microsoft AZURE** Classic and **Azure Resource Manager (ARM)**,GCP Deployed Web applications on Azure using Power Shell Workflow.
* Extensive experience and understanding in all phases of Software Development Life Cycle.
* Strong Analytical, Inter-personnel, Problem-solving, Communication skills and proven ability to rapidly acquire knowledge on new technologies and implement them in software projects.
* Experiences in using **build/deploy** tools such as Jenkins, **Helm chart creation,Docker and Open Shift** for Continuous Integration & Deployment for Microservices.
* Competent extent of skills on **DevOps** essential tools like Chef, Puppet, Ansible, **Docker, Git**, Jenkins, Maven.
* Responsible for on boarding application teams to build and deploy their code using **GitHub, Jenkins** and Ansible.
* Set-up **CI/CD** pipelines with **Jenkins** and hands on experience in building freestyle projects, pipeline, and multi branch pipeline Jobs through **Jenkins** file.
* Basic Knowledge of secure key storage facilities and techniques.
* Knowledge of asymmetric and symmetric cryptography.
* Hands on experience in **Manual& Automation** Testing, which involves User Interface, Functionality, Regression and System Integration Testing.
* Strong use of **Shell scripting** languages, including Bash for Linux and Mac platforms and Power Shell for Windows systems
* Strong understanding and working experience with **CI/CD** and available tools i.e., usage of Jenkins, Selenium, Cucumber, **Sonar and Gherkin**, component test automation.
* Facility with development methodologies, including **Agile.**
* Excellent configuration management using Puppet, **Chef and Ansible**.
* Proficient in writing **Jenkins** scripts for **CI/CD** Pipelines using **Groovy**.
* Wrote Ansible playbooks to launch **AWS** instances and used **Ansible** to manage web applications, configuration files, used mount points and packages.
* Experienced in full automation and configuration management desirable.
* Ample knowledge in using **Tomcat and JBOSS** Application servers for deployments
* Security and Compliance, e.g., **IAM** and cloud compliance/auditing/monitoring tools
* Strong experience in automating Vulnerability Management patching and CI/CD using Chef and other tools like GitLab, Jenkins, and **AWS/Open Stack**.
* Involved in very effective **Agile SCRUM** process & Automated Test Cases based on TDD methodology. Testing **REST API’s** using Advanced Rest Client.
* Well versed in Oracle RDBMS and **MY SQL** and proficient in **SQL queries.**
* Hands on Linux based systems, writing **Shell Scripts** and for prod and non-prod environments, CD Pipelines on Jenkins.
* Worked on JIRA Scrum Boards and Ticketing Tools, **Splunk** Logging for Issue debugging.
* Extensive experience in **NOSQL** databases like **MongoDB** and Elastic search.
* Expertise in **Scrum Methodologies** such as Agile and Waterfall methods, software development lifecycle management, continuous integration, build and release management and managed environments
* Involved in solving productions issues and critical remedies at client-side environment.
* Involvement in Code Review for the code units developed by the team.

**TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| **Operating Systems:** | Red Hat Linux, WINDOWS, Unix |
| **Cloud Tools and Technologies:** | Microsoft Azure, Amazon Web Services (AWS): EC2, S3, IAM, SNS, SQS, Lambda, server less, Cloud Watch, EKS, ECS, ECR, ELB, Auto Scaling, KMS, Cloud Front, Cloud Formation, Elastic File system, RDS, DMS, VPC, Direct Connect, Route 53, Cloud trail, Elastic Beanstalk. |
| **Languages:** | Java, XML, SHELL, RUBY, YAML, C#, .Net, Python |
| **Databases:** | MySQL, Oracle, Mongo DB |
| **Artifactory Repositories:** | Nexus |
| **CM Tools:** | Ansible, Chef, Puppet |
| **Web/Application Servers:** | Tomcat, JBoss |
| **Build Tools:** | Maven |
| **CI/CD Tools:** | Jenkins |
| **Automation container:** | Docker, Kubernetes, OpenShift |
| **SCM Tools:** | GIT, Bitbucket, Gitlab |
| **Bug Reporting Tools:** | JIRA |
| **Monitoring Tools:** | Splunk, AWS Cloud Watch, Prometheus, ELK- Elastic Search, Kibana |
| **Methodologies:** | Agile/Scrum, Waterfall |
|  |  |

**EDUCATION:**

* Masters –ComputerNetworking,Wichita State University,KS USA (**2014)**
* Bachelors –Electronics and Communication Engineering,GEC,Vijayawada, AP, India (**2012**)

**PROFESSIONAL EXPERIENCE:**

**Client: INFOVERITY, Dublin OH Jan 2019 – Till date**

**Role: Sr. Azure Lead / DevOps Engineer**

**Responsibilities:**

* Deploy, configure and maintain computation on Azure cloud.
* Troubleshoot issues and engage with teams and vendors for issue resolutions.
* Automate deployment and troubleshoot mechanism for quick service.
* Develop and implement new process and provide documentation of it.
* Handled work assignment through the ticketing system or form lead administrators.
* Performed software installation and maintenance task for cloud infrastructure.
* Conducted technical evaluation and proof of concept programs evaluating and implementing new technologies and tools.
* Configured Service Connections in **Azure DevOps** to connect to Azure Subscriptions using the App registration (Service Principle),powershell,SOC,**Helm chart creation** and Common library.
* Managed deployment to different environments using Jenkins and helped team by unblocking them resolving the pipeline issues.
* Involved in creating the **CD/CI** pipelines to execute the functional tests as part of the build process using Jenkins.
* Managed **GitHub** repositories and permissions, including branching and tagging.
* Create, maintain and administrating GitHub and Subversion repositories as source code managements.
* Implement patching solutions using a configuration management tool as well as leveraged update management feature in Azure and documented the end-to-end process to provide as a service.
* Resolving network issues, Provisioning Vnets, python and running testscripts and preparing network documentation.
* Worked extensively with **Helm chart creation**,powershell,**Chef,python,groovy,SOC, GCP, C#, Puppet**, **Salt & Ansible** in production environments.
* Worked on CI/CD pipelines Automation leveraging Jenkins as well quality analysis using SonarQube.
* Used Azure Monitoring for monitoring the daily activities performed in AZ cloud and setup alerts.
* Creating new build jobs, Integration testing jobs and deploy jobs in Jenkins to automate the process.
* Developed procedures to unify, streamline and automate application development and deployment procedures with Linux container technology using **Docker**.

**Environment:** Git, GitHub, GitLab, BitBucket, Jenkins, **python,groovy,SOC,powershell,** CI/CD, ArgoCD, Rancher, Maven, Python, Bash, Shell, Power shell, Unix, Linux, Docker, Dockerhub, Docker swarm, Docker Engine, Kubernetes, Kubernetes Cluster, Azure, Azure DevOps, AWS, Nexus, Jfrog, Terraform, Iaas, Paas, Saas, MySQL, DynamoDB, MongoDB, Prometheus, Grafana, Splunk, Nagios, Ansible, Chef, Puppet, Openshift.

**Client: MARKEL CORPORATION, GLEN ALLEN, VA Jan 2016 - Dec 2018**

**Role: Sr. Azure Lead / DevOps Engineer**

**Responsibilities:**

* Worked on **Microsoft Azure** (Public) Cloud to provide IaaS support to client as an Architect Level. Set up Automation to create Virtual Machines through Power Shell Script and Azure Portal, Created and managed Storage Account and Affinity Group in **Azure Portal**.
* Configured Site to Site **VPN** Connectivity. Configured Window Failover Cluster by creating Quorum for File sharing in **Azure Cloud**.
* Designed **AZURE Resource Manager (ARM)** templates and custom build steps using PowerShell. Identified and logged defects in VSTS and interacted with developers to prioritize the issues.
* Defined resources and specified deployment parameters to **Azure Resource Manager (ARM)** templates to create resource group of virtual machines.
* Create and maintain fully automated **CI/CD** pipelines for code deployment using groovy.
* Generated Docker files to build **Docker images, python,groovy,powershell, GCP, Helm chart creation, C#** and leveraged **Docker containers** in **Kubernetes**.
* Built and deployed **Docker containers** to break up monolithic app into microservices, improving developer workflow, increasing scalability, and optimizing speed.
* Reduced deployment time for critical agile project infrastructure from ~1 month to 2 days.
* Provided production and pre-prod support for multiple applications involving Java, python, GoLang involving tools like **Jenkins, Helm chart creation, Kubernetes** etc.
* Created **YAML** templates and updated packages of various applications using **Ansible Playbooks**.
* Tested procedures to unify streamline and automate applications development and deployment. Procedures with Red Hat **Linux container** technology using **Kubernetes and C#**.
* Used spring config server for centralized configuration and Splunk for centralized logging. Used Concourse and Jenkins for **Microservices** deployment.
* Consumed **REST services** in UI service layer with the help of http and resource providers.
* Regularly participated in Technical and Used Case review meetings for various functionalities in the application for python.
* Test, build, design, deployment, and ability to maintain continuous integration and continuous delivery process using tools like **Jenkins, GCP, mavenGit, C#** etc.
* Regularly participated in Technical and Used Case review meetings for various functionalities in the application.
* Involved in Interface Testing (**API testing**) for **Helm chart creation** and ensured data and control are working properly between middleware and backend systems using complex **Application Programming Interface** (API).

**Environment:** BitBucket, Jenkins, CI/CD, Maven, mavenGit , Bash, Shell, Power shell, Unix, Linux, Docker, Dockerhub, Docker swarm, Docker Engine, Kubernetes, Kubernetes Cluster, Helm chart, Azure, Azure DevOps, AWS, Terraform, Iaas, Paas, Saas, MySQL, DynamoDB, MongoDB, Prometheus, Grafana, Splunk, Ansible, Chef, Puppet, Openshift.

**Client:AETNA, Hartford CT. Aug 2013 - Dec 2015**

**Role: Sr. Cloud DevOps Engineer**

**Responsibilities:**

* Implemented a Continuous Delivery framework using **Azure pipelines**, Azure DevOps.
* Analyzed on premise applications infrastructure and designed VM sizing and disk sizes in Azure cloud.
* Configured Service Connections in **Azure DevOps** to connect to Azure Subscriptions using the App registration (Service Principle) and Common library.
* Designed, developed and built medium to large-scale pipelines using multiple DevOps tools like **groovy**, **JIRA and Selenium** is required.
* Involved in creating the **CD/CI** pipelines to execute the functional tests as part of the build process using Jenkins.
* Optimized current **CI/CD** process and streamline automation effort for server provisioning and deployment
* Installed and configured **Jenkins**, C# creation of master and slave nodes to handle multiple builds. Installation of plug-in wherever required for the implementation of those builds.
* Created **YAML** templates and updated packages of various applications using **Ansible Playbooks for GoLang**.
* Configured Jenkins pipeline for auto deployment through continuous integration (Jar build, AMI creation, Integration test, contract test, performance test, deploy in Test environment and then into Production.
* Configured **AWS Identity** and Access Management (IAM) to securely manage AWS users & groups and used policies & roles to allow or deny access to **AWS resources**.
* Worked on configuration and deployment of instances on AWS environment and familiar with EC2, Cloud watch, Elastic Search and managing security groups on AWS.
* Designed AWS Cloud Formation templates to create custom sized **VPC, subnets,GoLang,python NAT** to ensure successful deployment of Web applications and database templates.
* Designing and implementing for fully automated server build management, monitoring and deployment By Using DevOps Technologies like **Chef.**
* Responsible to develop and execute the functional test cases using **groovy and cucumber**.
* Consumed **REST services** in UI service layer with the help of http and resource providers.
* Regularly participated in Technical and Used Case review meetings for various functionalities in the application.
* Managed deployment to different environments using Jenkins and helped team by unblocking them resolving the pipeline issues.
* Involved in creating the **CD/CI** pipelines to execute the functional tests as part of the build process using Jenkins.
* Worked on container orchestration tool Docker and deployment tool as Kubernetes. Defined Docker File for application image and **YAML files** for deployment.
* Involved in Interface Testing (**API testing**) and ensured data and control are working properly between middleware and backend systems using complex Application Programming Interface (**API**).

**Environment:** Git, GitHub, GitLab, BitBucket, Jenkins, CI/CD, Maven, Python, Bash, Shell, Power shell, Unix, Linux, Docker, Dockerhub, Docker swarm, Docker Engine, Kubernetes, Azure, AWS, Nexus, Jfrog, Terraform, MongoDB, Splunk, Ansible, Chef, Puppet, Openshift.